University of Tokyo/RICOH at NTCIR-3 Web Retrieval Task

Masashi TOYODA and Masaru KITSUREGAWA
Institute of Industrial Science, University of Tokyo
4-6-1 Komaba, Meguro-ku, Tokyo 153-8505, JAPAN
{toyoda,kitsure}@tkl.iis.u-tokyo.ac.jp

Hiroko MANO, Hideo ITOH and Yasushi OGAWA
Software R&D Group, RICOH Co., Ltd.
1-1-17 Koishikawa, Bunkyo-ku, Tokyo 112-0002, JAPAN
{mano,hideo,yogawa}@src.ricoh.co.jp

Abstract

In NTCIR-3 Web Task, we introduced new approaches in (1) similarity retrieval using one known relevant document and pseudo-relevance feedback and (2) topic and target retrieval incorporating link analysis. The experiments showed that both approaches were promising.

Keywords: *NTCIR, Web retrieval, query expansion, link analysis*

1 Introduction

For NTCIR-3 Web Task, the University of Tokyo/RICOH group submitted runs in subtasks I-A1 and II-A1 (survey-topic retrieval), I-A2 and II-A2 (survey-similarity retrieval) and I-B and II-B (target retrieval), using both the 10G and 100G data sets.

Our main focuses at NTCIR-3 Web Task were:

1) to evaluate our strategy of using pseudo-relevance feedback in similarity retrieval given one known relevant document

2) to test our new approach incorporating link analysis based on Kleinberg's HITS [9] in topic retrieval and target retrieval using the 100G data set

2 System

The system consists of two components, a search engine FTS which handles document retrieval based solely on the content, and a link analyzer using a modified version of the HITS algorithm, Companion– [17], which extracts authoritative pages on a given topic from the structure of the Web graph.

The search engine retrieves and ranks a set of documents from the document collection using its content-based algorithm. Starting from the retrieved set, the link analyzer then examines the links, rates the authority of each document and re-rank the retrieved set based on the link analysis result.

In what follows, we describe each of the ranking methods in more detail and discuss what it yielded as results.

3 Ranking based on content

3.1 Search engine FTS

In all the runs, documents that match the topic in terms of the content were retrieved by the search engine FTS, which was also used for NTCIR-2 by Ricoh [10].

The basic features of the system are:

- Effective and robust document ranking based on the probabilistic model [15] with query expansion using pseudo-relevance feedback [11]
- Scalable and efficient indexing and search based on the inverted file module [10]
- Hybrid retrieval combining n-gram indexing and word-based query processing using an originally developed Japanese morphological analyzer

For NTCIR-3, we added a more sophisticated query processing mechanism that allows finer control on query terms, including phrasal forms, as well as various speed-up measures to improve overall efficiency.

In the following, the methods used and their results for content-based retrieval are discussed for each subtask.

3.2 Survey-topic retrieval I-A1 and II-A1

For the survey-topic retrieval subtasks I-A1 and II-A1, we submitted two and four mandatory runs, respectively, using only content-based methods.

3.2.1 Methods

The outline of retrieval is as follows.

1. Query term extraction
   Input query string is transformed into a sequence of words using the Japanese morphological analyzer. Query terms are extracted by matching the sequence against the patterns that define combinations of terms appropriate as query terms, expressed in regular expression on each word form or part-of-speech tag assigned by the analyzer. Stop words are eliminated using a stop word dictionary. For initial retrieval, both "single terms" and "phrasal terms" are used. A phrasal term consists of two adjacent words in the query string.

2. Initial retrieval
   Each query term is assigned a weight w_t, and documents are ranked according to the score $s_{q,d}$ as follows:

   $$w_t = \log\left(k'_4 \cdot \frac{N}{n_t} + 1\right),$$

   $$\begin{aligned} s_{q,d} &= \textstyle\sum_{t \in q} \frac{f_{t,d}}{K+f_{t,d}} \cdot \frac{w_t}{k'_4 \cdot N+1}, \\ K &= k_1\left((1-b) + b\frac{l_d}{l_{ave}}\right), \end{aligned}$$

   where N is the number of documents in the collection, n_t is the document frequency of the term t, $f_{t,d}$ is the in-document frequency of the term, l_d is the document length, l_{ave} is the average document length, and k'_4, k_1 and b are parameters.

   Weights for phrasal terms are set lower than those for single terms.

3. Seed document selection
   As a result of the initial retrieval, top-ranked documents are assumed to be relevant (pseudo-relevant) to the query and selected as a "seed" of query expansion.

4. Query expansion
   Candidates of expansion terms are extracted from the seed documents by pattern matching as in the query term extraction mentioned above.

   The candidates are ranked on the Robertson's Selection Value [13], or RSV_t, and top-ranked terms are selected as expansion terms. The weight is re-calculated as $w2_t$ with the Robertson/Sparck-Jones formula [14]

   $$RSV_t = w2_t \cdot (r_t/R - n_t/N),$$

   $$w2_t = \alpha \cdot w_t + (1-\alpha) \cdot \log \frac{\frac{r_t+0.5}{R-r_t+0.5}}{\frac{n_t-r_t+0.5}{N-n_t-R+r_t+0.5}},$$

   where R is the number of relevant documents, r_t is the number of relevant documents containing the term t and α is a parameter.

   Phrasal terms are not used for query expansion because phrasal terms may be too specific for use with pseudo-relevance feedback.

   The weight of initial query term is re-calculated with the same formula as above, but with a different α value and an additional adjustment to make the weight higher than expansion terms.

5. Final retrieval
   Using the initial query terms and expansion terms, the ranking module performs second retrieval to produce the final results.

We used the data sets de-tagged by NII.

3.2.2 Results and discussion

The evaluation results of our submitted runs are summarized in Table 1 for the 100G data set and Table 2 for the 10G data set[1], where we used the qrels data on the content-only judgment and the documents judged to be "H" or "A" were taken as relevant ones. For comparison purposes, comparable unsubmitted runs are also included.

Type	AveP	P@10	P@20	Run-ID
tn	0.1211	0.1809	0.1745	–
te	0.1506	0.2213	0.1968	LA1-1
dn	0.1318	0.2085	0.1851	–
de	0.1548	0.2340	0.2138	LA1-3

tn: title only, without query expansion
te: title only, with query expansion
dn: desc only, without query expansion
de: desc only, with query expansion

Table 1. Evaluation results of I-A1

[1] Since all our runs have run-IDs that start with 'GRACE,' we omit 'GRACE' from the run-IDs.

Type	AveP	P@10	P@20	Run-ID
tn	0.2148	0.1756	0.1367	SA1-1
te	0.2260	0.1822	0.1444	SA1-2
dn	0.2058	0.1644	0.1356	SA1-3
de	0.2365	0.1778	0.1444	SA1-4

Table 2. Evaluation results of II-A1

Table 1 and Table 2 indicate that our query expansion using pseudo-relevance feedback contributed in improving average precision by as much as 24%. As for title-only vs. desc-only comparison, when query expansion was applied, the desc-only runs yielded better average precision than the corresponding title-only runs for both data sets, but without query expansion, that does not always hold true.

3.3 Survey-similarity retrieval I-A2 and II-A2

We submitted four mandatory runs for each of the similarity retrieval subtasks I-A2 and II-A2, using only content-based methods.

3.3.1 Methods

Our approach was focused on the relevance feedback technique, in which the known relevant document rdoc[1] was used as one of the seed documents for query expansion, rather than as part of a query. Because the relevance information was given by just one relevant document rdoc[1] in the mandatory run, we compensated for lack of relevance information by adding pseudo-relevance information.

The retrieval process is outlined as follows (See previous sections for more detail):

1. Initial retrieval is performed for the title field of each topic.
2. Query expansion is performed using rdoc[1] and the top-ranked documents (pseudo-relevant documents) in the initial retrieval.
3. Final retrieval is performed for the expanded query.

Another strategy we employed was to duplicate the rdoc[1] in the seed (i.e., rdoc[1] was given twice) so that the positive influence expected from the relevant document would be enhanced.

3.3.2 Results and discussion

The evaluation results of our submitted runs are summarized in Table 3 and Table 4.

Model	AveP	P@10	P@20	Run-ID
d-0	0.1913	0.3000	0.2660	LA2-1
d-1	0.1966	0.3000	0.2745	LA2-2
d-3	0.1977	0.2766	0.2713	LA2-3
s-3	0.1769	0.2468	0.2330	LA2-4
baseline	0.1211	0.1808	0.1745	–

Table 3. Evaluation results of I-A2

Model	AveP	P@10	P@20	Run-ID
d-0	0.2754	0.2022	0.1478	SA2-1
d-1	0.2935	0.2289	0.1622	SA2-2
d-3	0.2905	0.2311	0.1633	SA2-3
s-3	0.2465	0.2133	0.1656	SA2-4
baseline	0.2148	0.1756	0.1367	–

Table 4. Evaluation results of II-A2

In the tables, the model "d-n" means that the relevant document rdoc[1] is used in duplicate and n pseudo-relevant documents are used for query expansion. The model "s-n" means that rdoc[1] is used without duplication and n pseudo-relevant documents are used.

As a baseline, we show the evaluation results of the runs produced using only the title field without query expansion. Comparing each model with the baseline, query expansion using relevance feedback produced a large positive effect in average precision and top 10 and top 20 precision.

Comparing the results of d-3 and s-3, the duplication of rdoc[1] increased retrieval performance in both the subtasks I-A2 and II-A2.

Comparing d-0 with d-1 and d-3, the blending relevant and pseudo-relevant documents increased retrieval performance in both subtasks. However, the number of pseudo-relevant documents which resulted in the best performance is different in the subtask I-A2 and II-A2.

We conclude that blending pseudo-relevance information and little relevance information enhanced by duplication produces better retrieval performance.

3.4 Target retrieval I-B and II-B

For the target retrieval subtasks I-B and II-B, we submitted two and four mandatory runs, respectively, using only content-based methods.

3.4.1 Methods

The same procedure as described in the survey-topic retrieval section was used.

3.4.2 Results and discussion

The evaluation results of our submitted runs are summarized in Table 5 and Table 6. Note that, since the same query set was used for evaluation in I-A1 and I-B, and in II-A1 and II-B, the P@10 values and P@20 values are the same respectively between these subtasks.

Type	P@10	P@20	Run-ID
tn	0.1809	0.1745	–
te	0.2213	0.1957	LB-1
dn	0.2085	0.1851	–
de	0.2340	0.2138	LB-3

Table 5. Evaluation results of I-B

Type	P@10	P@20	Run-ID
tn	0.1756	0.1367	SB-1
te	0.1822	0.1444	SB-2
dn	0.1644	0.1356	SB-3
de	0.1778	0.1444	SB-4

Table 6. Evaluation results of II-B

As in the survey retrieval subtask, using query expansion was effective in the target retrieval as well, where top 20 ranking counts. When we compare the title-only runs with the desc-only runs, we observe the better performance of the desc-only runs for the 100G data set, but no significant difference for the 10G data set.

4 Improving ranking with link analysis

In past TREC Web tracks [8, 6, 7], many groups tried to incorporate various link analysis techniques, including Kleinberg's HITS [9], and Larry Page and Surgey Brin's PageRank [12]. However, in most cases, link analysis provided limited or negative effect in topic query tasks. In this section, we examine whether link analysis can improve retrieval effectiveness on the survey-topic retrieval subtask I–A1 and the target retrieval subtask I–B.

We use our modified version of HITS algorithm, Companion– [17], and test two blending methods. Results show modest improvements from the baseline FTS results. We also performed experiments using a larger link data in Kitsuregawa Laboratory, University of Tokyo (250GB, 40M pages, crawled in early October, 2001) to investigate whether the size of link data has effect on retrieval results.

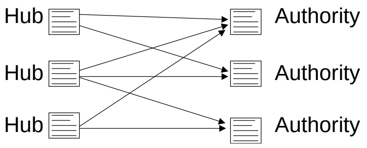

Figure 1. Typical graph structure of hubs and authorities

4.1 Method

Some of past TREC participants attempted to exploit HITS such as [5, 16, 2, 4]. Our method is also based on HITS, which extracts related pages to a given topic with the notion of *authorities* and *hubs*. An authority is a page with good contents on a topic, and is pointed to by many good hub pages. A hub is a page with a list of hyperlinks to valuable pages on the topic, that is, points to many good authorities. HITS is an algorithm that extracts authorities and hubs from a subgraph of the Web, built from result pages by a search engine and adjacent pages. Figure 1 shows a typical graph structure extracted by HITS. As shown in the graph, HITS extracts frequently co-cited pages as authorities.

In the following, we first explain the Companion– algorithm, then describe our blending methods.

4.1.1 Companion–

Companion– [17] takes a seed page as input, then outputs related pages to the seed. It first builds a subgraph of the Web around the seed, and extracts authorities and hubs in the graph using HITS [9]. Then authorities are returned as related pages. Companion– uses a subgraph narrower than HITS and its alternative Companion [3]. As a result, Companion– gave better results than HITS and Companion in most cases, and was outstanding at top 10 precision. For more details, please refer [17].

Companion– can be applied to multiple seeds without any change. In the following, we describe the process of Companion– with multiple seeds.

First, it builds a vicinity graph of given seeds, which is a subgraph of the Web around the seeds. A vicinity graph is a directed graph, (V, E), where nodes in V represent Web pages, and edges in E represent links between these pages. As shown in Figure 2, V consists of the seeds, a set of nodes pointing to the seeds (B), and another set of nodes pointed to by nodes in B. When following outgoing links from each node in B, the order of links in the

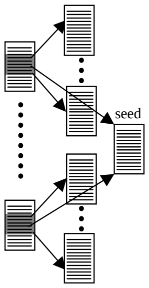

Figure 2. Vicinity graph

node is considered. Not all the links are followed but only R links immediately preceding the link pointing to each seed, and R links immediately succeeding the link. This is based on an observation that links to related pages are gathered in a small portion of a page.

When some pages written by the same author are pointing to a page p, p is improperly considered as an authoritative page. For decreasing such influence of a single author, it assigns two kinds of weights, an *authority weight* and a *hub weight* to each edge. The authority weight is used for calculating an authority score of each node, and the hub weight is used for calculating a hub score of each node. Companion– uses the following weighting method proposed by Bharat and Henzinger [1]. For simplicity, we consider that pages in a same server are written by the same author.

- If two nodes of an edge are in the same server, the edge has the value 0 for both weights.
- If a node has n incoming edges from the same server, the authority weight of each edge is $1/n$.
- If a node has m outgoing edges to the same server, the hub weight of each edge is $1/m$.

Then it calculates a hub score, $hub(n)$, and an authority score, $auth(n)$ for each node n in the vicinity graph, (V, E). The following is the process of the calculation, where $auth\_wt(n, m)$ and $hub\_wt(n, m)$ represent the authority weight and the hub weight of the edge from n to m, respectively.

Step 1. Initialize $hub(n)$ and $auth(n)$ of each node n to 1.

Step 2. Repeat the following calculation until $hub(n)$ and $auth(n)$ have converged for each node n.
For all node n in V,

$$hub(n) \leftarrow \sum_{(n,m)\in E} auth(m) \cdot hub\_wt(n,m)$$

For all node n in V,

$$auth(n) \leftarrow \sum_{(m,n)\in E} hub(m) \cdot auth\_wt(m,n)$$

Normalize $hub(n)$, so that the sum of squares to be 1.
Normalize $auth(n)$, so that the sum of squares to be 1.

Step 3. Choose nodes with positive authority scores as results.

4.1.2 Blending scores

We tested two methods for blending FTS scores and authority scores, and submitted results of the best method based on the dry-run evaluation. Our blending methods take a ranked list of top 1000 documents (R_{1000}) returned from the FTS engine, and perform re-ranking using Companion–. The following is the detailed process of the blending method. The two methods (a) and (b) differ only on blending functions in the step 3.

Step 1. Extract top N results R_N from R_{1000}, and apply Companion– to R_N.

Step 2. Choose pages that have positive authority scores, and that are included in R_{1000}. We call a set of these authoritative pages A. From its definition, $A \subset R_{1000}$.

Step 3. (a) Calculate the score sc_a of each page as follows.

$$sc_a(p) = \frac{(1-\alpha)\cdot fts(p)}{\max_{q\in R_{1000}} fts(q)} + \frac{\alpha \cdot auth(p)}{\max_{r\in A} auth(r)}$$

(b) Give each page in A a constant bonus score based on the maximum score given by FTS. A new score sc_b of a page p in A becomes as follows.

$$sc_b(p) = fts(p) + \beta \cdot \max_{q\in R_{1000}} fts(q),$$

where $fts(p)$ is a score given by FTS, $auth(p)$ is an authority score given by Companion–. α and β are constants to control the effect of link analysis.

The first method (a) directly blends FTS scores and authority scores. In our experiments with the dry-run results, we found that the authority score of a page represents its importance or popularity of the page in some aspect, but they are sometimes independent to the query topic. Therefore, when we directly blend authority scores and FTS scores, nonrelevant pages may have higher scores than relevant pages.

The second method (b) solves this problem by giving a constant bonus score to authoritative pages extracted by Companion−. In this way, we put emphasis on authoritative pages, preserving their order in the FTS result.

We submitted only results of the method (b), since the method (b) provides better results than (a) in our experiments based on the dry-run evaluations. We compared two method based on the formal-run evaluation in Section 4.3.

4.2 Survey-topic retrieval I-A1 and Target retrieval I-B

For survey-topic retrieval I-A1, we submitted two official runs LA1-2 and LA1-4. LA1-2 is based on the title-only result of FTS with query expansion (an unsubmitted run different from LA1-1, which used a slightly different set of parameter values). LA1-4 is based on the desc-only result of FTS with query expansion (LA1-3). Both LA1-2 and LA1-4 use the link data in NTC 100GB dataset. In these official runs, we use the method (b), and chose parameters for blending (See Section 4.1.2) as follows: $N = 100$; and $\beta = 0.06$. These parameters were determined by the dry-run results, and by our own evaluation on 20 formal queries.

We also submitted two unofficial runs LA1-6 and LA1-8 based on the unsubmitted title-only run mentioned above and LA1-3, respectively. These two runs use a link data of a Web archive in Kitsuregawa Laboratory, University of Tokyo. This link data ("Kilab data" in the following) was built from a 250GB Web archive with 40M pages, crawled in early October, 2001. In these unofficial runs, we chose parameters for blending as follows: $N = 100$; $\beta = 0.14$ (for LA1-6); and $\beta = 0.13$ (for LA1-8). These parameters were also determined by our experiments.

In the same way, we submitted four runs LB-2, LB-4, LB-6, and LB-8 for the target retrieval subtask I-B.

Table 7 and 8 show evaluation results of the survey-topic and target retrieval subtasks with the qrels data. Modest improvements are shown in mean average precision, and top 20 precision for all cases. In some cases, top 10 precision with the NTC data decreases from baselines, but with the Kilab data it increases in all cases. We obtained similar results with the qprels data.

In most cases, a larger link data provides better results; we see that all results with the Kilab data are better than those with the NTC data. However, the NTC data has only a marginal effect on results. In other words, the link data inside the 100GB dataset is insufficient for improving retrieval effectiveness, and links outside the dataset can provide additional performance.

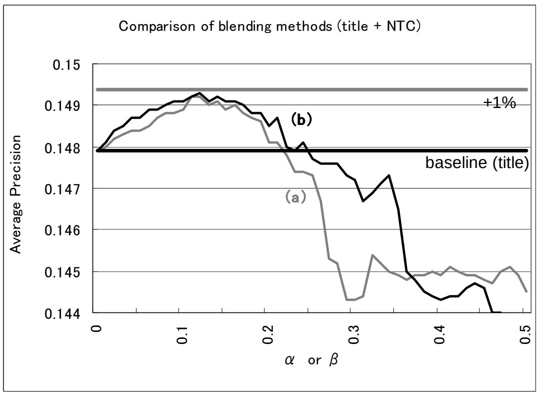

Figure 3. Comparison of blending methods (title + NTC data)

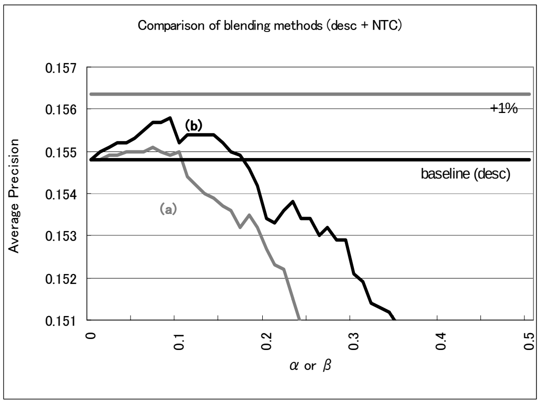

Figure 4. Comparison of blending methods (desc + NTC data)

4.3 Comparing blending methods

We also compared two blending methods in Section 4.1.2 using the formal-run evaluations. Figure 3 and Figure 4 show mean average precision of each method as the function of the parameter α or β. As the baseline, we use the title-only result of FTS in Figure 3, and use the desc-only result in Figure 4. In both figures, we use the NTC data for link analysis.

In both cases, the method (b) is better than the method (a), but differences are not so significant. In the title-only configuration (Figure 3), the maximum gain is about 1% from the baseline in both methods, and the method (b) is slightly better than the method (a). The advantage of the

		content-only			with-link		
Configuration	Run-ID	AveP	P@10	P@20	AveP	P@10	P@20
title (baseline)	–	0.1479	0.2149	0.1979	0.1537	0.2489	0.2457
title + NTC data	LA1-2	0.1489	0.2106	0.2011	0.1548	0.2447	0.2521
title + Kilab data	LA1-6	0.1501	0.2213	0.2032	0.1592	0.2660	0.2543
desc (baseline)	LA1-3	0.1548	0.2340	0.2138	0.1479	0.2681	0.2457
desc + NTC data	LA1-4	0.1555	0.2340	0.2170	0.1488	0.2681	0.2511
desc + Kilab data	LA1-8	0.1552	0.2362	0.2191	0.1498	0.2745	0.2628

Table 7. Evaluation results of I-A1 (with link analysis)

		content-only		with-link	
Configuration	Run-ID	P@10	P@20	P@10	P@20
title (baseline)	–	0.2149	0.1979	0.2489	0.2457
title + NTC data	LB-2	0.2106	0.2011	0.2447	0.2521
title + Kilab data	LB-6	0.2213	0.2032	0.2660	0.2543
desc (baseline)	LB-3	0.2340	0.2138	0.2681	0.2457
desc + NTC data	LB-4	0.2340	0.2170	0.2681	0.2511
desc + Kilab data	LB-8	0.2362	0.2191	0.2745	0.2628

Table 8. Evaluation results of IB (with link analysis)

method (b) is greater in the desc-only configuration (Figure 4). The maximum gain of the method (b) is about 0.6%, and that of the method (a) is about 0.1%.

The best value of the parameter α or β is around 0.1 in both cases. Currently, we select these parameters by preliminary evaluations. Automatic determination of parameters is future work.

5 Conclusions

The experiments showed that, in similarity retrieval, our strategy using pseudo-relevance feedback combined with relevant document duplication was very effective. Also from the experiments, we found that the approach incorporating link analysis based on our modified version of the HITS algorithm resulted in positive gains, especially when a larger link data was used.

References

[1] K. Bharat and M. Henzinger. Improved Algorithms for Topic Distillation in a Hyperlinked Environment. In *Proceedings of the 21th Annual International ACM SIGIR Conference (SIGIR '98)*, 1998.

[2] F. Crivellari and M. Melucci. Web Document Retrieval using Passage Retrieval, Connectivity Information, and Automatic Link Weighting – TREC-9 Report. In *The Ninth Text REtrieval Conference (TREC-9)*, 2001.

[3] J. Dean and M. R. Henzinger. Finding related pages in the World Wide Web. In *Proceedings of the 8th World-Wide Web Conference*, 1999.

[4] J. Gevrey and S. M. Rüger. Link-based Approaches for Text Retrieval. In *The Tenth Text REtrieval Conference (TREC-2001)*, 2002.

[5] C. Gurrin and A. F. Smeaton. Dublin City University Experiments in Connectivity Analysis for TREC-9. In *The Ninth Text REtrieval Conference (TREC-9)*, 2001.

[6] D. Hawking. Overview of the TREC-9 Web Track. In *The Ninth Text REtrieval Conference (TREC-9)*, 2001.

[7] D. Hawking and N. Craswell. Overview of the TREC-2001 Web Track. In *The Tenth Text REtrieval Conference (TREC-2001)*, 2002.

[8] D. Hawking, E. Voorhees, N. Craswell, and P. Bailey. Overview of the TREC-8 Web Track. In *The Eighth Text REtrieval Conference (TREC-8)*, 2000.

[9] J. M. Kleinberg. Authoritative Sources in a Hyperlinked Environment. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, 1998.

[10] Y. Ogawa and H. Mano. RICOH at NTCIR-2. In *Proceedings of the Second NTCIR Workshop Meeting*, pages 121–123, 2001.

[11] Y. Ogawa, H. Mano, M. Narita, and S. Honma. Structuring and expanding queries in the probabilistic model. In *The Eighth Text REtrieval Conference (TREC-8)*, pages 541–548, 2000.

[12] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the Web. Technical report, Stanford Digital Library Technologies Project, 1998.

[13] S. E. Robertson. On term selection for query expansion. *Journal of Documentation*, 46(4):359–364, 1990.

[14] S. E. Robertson and K. Sparck-Jones. Relevance weighting of search terms. *Journal of ASIS*, 27:129–146, 1976.

[15] S. E. Robertson and S. Walker. On relevance weights with little relevance information. In *Proceedings of the 20th Annual International ACM SIGIR Conference (SIGIR '97)*, pages 16–24, 1997.

[16] J. Savoy and Y. Rasolofo. Report on the TREC-9 Experiment: Link-Based Retrieval and Distributed Collections. In *The Ninth Text REtrieval Conference (TREC-9)*, 2001.

[17] M. Toyoda and M. Kitsuregawa. Creating a Web Community Chart for Navigating Related Communities. In *Conference Proceedings of Hypertext 2001*, pages 103–112, 2001.